

---

# LeadForge, Explained

A B2B lead-enrichment API — in plain English, then in nerd-speak.

## Part 1 — In Plain English

### The 10-second version

You give LeadForge a list of company websites. It tells you who works there, what those people do, how to reach them, and what the company actually sells. That's it.

### Why anyone would want that

Imagine you sell software to dentists. You have a spreadsheet of 500 dental-clinic websites — but no names, no emails, no phone numbers, no idea which clinic is a single-office practice vs. a 40-location chain. Cold-calling 500 generic info@ addresses is a waste of a week.

LeadForge takes that list of 500 websites and hands you back a clean spreadsheet: owner's name, role, work email, LinkedIn, company size, what services they offer, and which clinics actually look like good prospects. What used to take a virtual assistant a week now takes about 90 seconds.

### Who uses it

- **Salespeople** — turn website lists into call lists
- **Recruiters** — find hiring managers at companies they're prospecting
- **Marketers** — segment company lists by size, industry, tech stack
- **Founders** — research customers + competitors at scale
- **Agencies** — deliver lead lists to clients without doing it by hand

### How it costs

Tools like Clearbit and ZoomInfo charge \$1,000–\$1,500 a year (or much more) for the same kind of data. LeadForge is **\$97 once**. You pay one time, you get a key, you keep the key for life. No subscriptions, no seat fees, no per-lead pricing.

### How a human actually uses it

Most buyers don't write code. They give their LeadForge key to: a virtual assistant, a Make.com or Zapier flow, a junior marketer, or ChatGPT/Claude (which can call the API for them). Paste in websites, get back enriched leads, drop them into a CRM or an email tool. That's the whole loop.

*One-line pitch: "LeadForge turns a list of company websites into a list of decision-makers you can actually contact — for a one-time \$97 instead of \$1,200 a year."*

---

## Part 2 — For the Tech Nerds

### TL;DR

LeadForge is a REST API that performs domain-keyed B2B contact + firmographic enrichment plus a multi-step workflow runner. One-time license, lifetime key, rate-limited per-key, batched up to 50 domains per call.

### Stack

FastAPI (Python 3.x) on Render • SQLite on persistent disk • Stripe Checkout for billing • Gmail SMTP for delivery • token-gated admin endpoints • signed Stripe webhooks for refund-driven key deactivation. Synchronous fulfillment on the success page, webhook-based refund auto-revoke. Live at [leadforge-rh03.onrender.com](https://leadforge-rh03.onrender.com).

### Endpoints (excerpt)

POST	/api/enrich	Bulk domain → contact + firmographic record (≤50/call)
POST	/api/workflow/run	Chained enrichment / scoring / dedup steps (≤20 steps)
GET	/api/usage	Per-key request counters + quota state
GET	/buy/{slug}	Generic Stripe Checkout redirect (yaml-driven)
GET	/success/{slug}	Synchronous fulfillment + thank-you page
POST	/webhook/stripe	checkout.session.completed + charge.refunded routing

### Auth model

Bearer token via `Authorization: Bearer <api_key>`. Keys are issued one-per-Stripe-session, idempotent on `session_id`, deactivatable via signed `charge.refunded` webhook events or the admin override endpoint. No per-call billing — flat-rate license.

### Example call

```
curl -X POST https://leadforge-rh03.onrender.com/api/enrich \
  -H 'Authorization: Bearer lf_live_xxx' \
  -H 'Content-Type: application/json' \
  -d '{"domains":["acme.com","globex.com"]}'
```

### What you get back (shape)

Per-domain JSON: company name, industry, employee-count band, headquarters, tech-stack signals, primary contacts (name, title, email, LinkedIn), confidence scores, and a normalized lead-fit score. Errors are per-row, not per-batch — partial responses are the norm.

### Why a one-time fee works

Flat-rate licensing eliminates the bookkeeping overhead of seat counts and per-call metering. Margin comes from the data sources being amortized across the buyer base, and from rate-limiting + per-key request caps that prevent abuse. Buyers self-select into 'I will use this enough to justify \$97' — exactly the cohort that produces support-light, retention-stable customers.

### Architecture footnote

Product registry now driven by a single `products.yaml` + generic `/buy/{slug}` + `/success/{slug}` routes. Adding a new digital product is a yaml entry plus an env var — no code changes. Webhook gating uses `price-id` → product lookup so each product's fulfillment runs only for its own line items.